



# Calcul parallèle pour l'interaction fluide-structure en situation explosive

Vincent Faucher

## ► To cite this version:

Vincent Faucher. Calcul parallèle pour l'interaction fluide-structure en situation explosive. 10e colloque national en calcul des structures, May 2011, Giens, France. pp.Clé USB. hal-00592707

**HAL Id: hal-00592707**

**<https://hal.science/hal-00592707>**

Submitted on 3 May 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Calcul parallèle pour l'interaction fluide-structure en situation explosive

V. Faucher<sup>1</sup>

<sup>1</sup> CEA, DEN, DM2S, SEMT, DYN, France, [vincent.faucher@cea.fr](mailto:vincent.faucher@cea.fr)

**Résumé** — La présente contribution propose de faire une revue des problématiques de recherche et développement dans le domaine du calcul parallèle en dynamique rapide avec couplage de modélisations et interaction fluide structure. Sont en particulier présentées les questions relatives à la gestion générique des connexions cinématiques par multiplicateurs de Lagrange, ainsi que la concurrence des différents algorithmes au niveau de la résolution parallèle. Les recherches en cours et les solutions proposées sont illustrées sur des applications industrielles avec le code EUROPLEXUS.

**Mots clefs** — Dynamique rapide explicite, IFS, connexions cinématiques.

## 1 Eléments de contexte

La simulation des transitoires brutaux issus de situations accidentelles requiert fréquemment la modélisation simultanée de fluides (liquides ou gazeux), en interaction avec des structures pouvant se déformer jusqu'à la ruine. Ceci conduit à coupler des modélisations différentes, telles que des éléments finis, des volumes finis ou des particules SPH pour le fluide, et des éléments finis volumiques ou surfaciques, des particules SPH ou des éléments discrets pour la structure (cf. figure 1). Des exemples de tels couplages en interaction fluide-structure figurent dans [1][2].

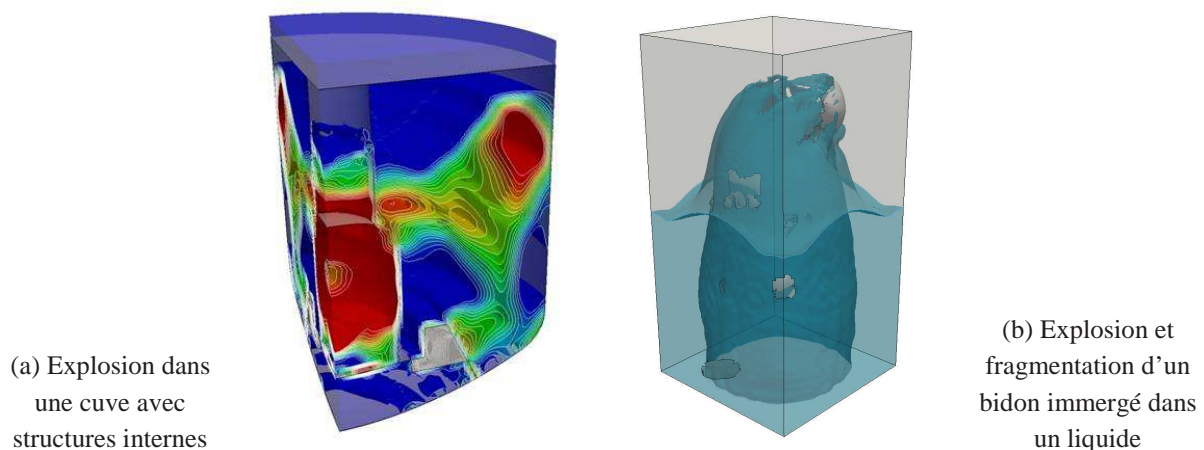


FIG. 1 : Interaction fluide-structure en situation explosive

La question des connexions cinématiques entre les différentes modélisations est fondamentale et le choix est fait dans EUROPLEXUS de vérifier les équations de liaisons de manière exacte au moyen de multiplicateurs de Lagrange, sans recourir à des paramètres de pénalisation à fixer par l'utilisateur. Au niveau algorithmique, ceci introduit une spécificité forte au niveau de la résolution temporelle : le schéma d'intégration est classiquement le schéma des différences centrées, mais la résolution d'un système linéaire est nécessaire pour le calcul des forces de connexion, ce qui produit un algorithme de

résolution partiellement implicite.

La gestion en parallèle de l'assemblage et de la résolution du problème posé sur les multiplicateurs de Lagrange pour l'ensemble des connexions autorisées par le programme entre en concurrence avec le parallélisme classique des codes de calcul en dynamique rapide et fait l'objet du présent document.

## 2 Algorithme général

### 2.1 Equilibre dynamique en interaction fluide-structure avec intégration explicite en temps

D'une manière générale, la conservation de la quantité de mouvement à l'instant  $t^{n+1}$  du processus d'intégration en temps, toutes les quantités étant supposées connues aux instants  $t^n$  (déplacements, pour la structure uniquement) et  $t^{n+1/2}$  (vitesses), se présente sous la forme :

$$\begin{aligned} \mathbf{M}^{n+1} \ddot{\mathbf{U}}^{n+1} + \mathbf{F}_i(\mathbf{U}^n, \dot{\mathbf{U}}^{n+1/2}) + \mathbf{C}^{n+1T} \boldsymbol{\Lambda}^{n+1} &= \mathbf{F}_{ext}^{n+1} \\ \mathbf{C}^{n+1} \dot{\mathbf{U}}^{n+3/2} &= \mathbf{B}^{n+1} \\ \dot{\mathbf{U}}^{n+3/2} &= \dot{\mathbf{U}}^{n+1/2} + \Delta t \cdot \ddot{\mathbf{U}}^{n+1} \end{aligned} \quad [\text{Eq. 1}]$$

Les forces  $\mathbf{F}_i$  représentent les forces internes pour la structure ou les forces de transport pour le fluide. Le système est complété pour le fluide par les équations de conservation de la masse et de l'énergie, traitées dans un formalisme volume fini, en calculant les flux des quantités à travers les faces de cellules du maillage fluide.  $\mathbf{M}^{n+1}$  est une matrice diagonale.

La matrice  $\mathbf{C}^{n+1}$  est la matrice des connexions cinématiques liant les vitesses des nœuds du fluide et de la structure, dont l'écriture est présentée au paragraphe suivant. Sa dimension, son profil et ses coefficients sont variables au cours du temps.

Au préalable au calcul des accélérations nodales permettant le passage à l'instant suivant, les multiplicateurs de Lagrange sont obtenus par condensation à partir d'un système de la forme :

$$\mathbf{C}^{n+1} \mathbf{M}^{n+1-1} \mathbf{C}^{n+1T} \boldsymbol{\Lambda}^{n+1} = \mathbf{H}^{n+1} \boldsymbol{\Lambda}^{n+1} = \mathbf{S}^{n+1} \quad [\text{Eq. 2}]$$

### 2.2 Ecriture des liaisons d'interaction fluide-structure

Deux types de liaisons fluide-structure interviennent classiquement pour la simulation d'un transitoire explosif [1][3]. Des liaisons conformes nœud à nœud, lorsque la structure représente l'enveloppe du domaine fluide, ou des liaisons diffuses pour les structures immergées, pour lesquelles disposer d'un maillage conforme conduit à des coûts importants pour la mise en donnée et à un manque de robustesse des algorithmes de mise à jour ALE de la grille fluide. La figure 2 illustre les deux types de liaisons.

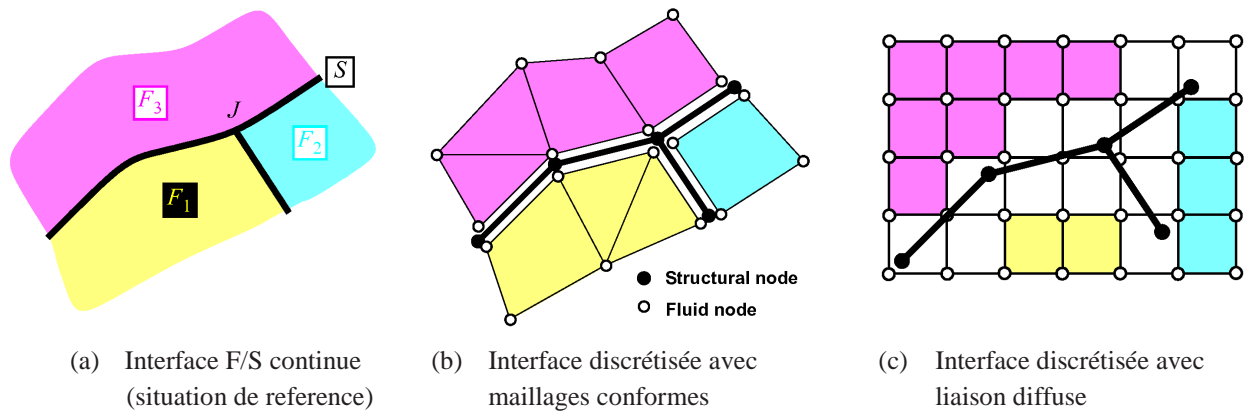


FIG. 2 - Liaisons d'interaction fluide-structure

L'interaction avec maillage conforme produit des liaisons permanentes, dont les coefficients sont variables avec la direction normale à la structure. L'interaction diffuse avec maillages déconnectés topologiquement génère des liaisons dont le support et les coefficients varient au cours du temps.

Le second type de liaison est indispensable pour traiter des problèmes où intervient la fragmentation de la structure et l'écoulement à travers les ouvertures créées, comme l'indique la figure 1-b ou la figure 3.

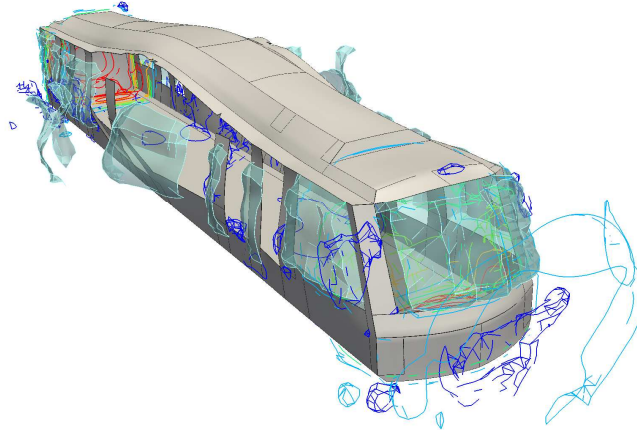


FIG. 3 – Simulation d'une explosion dans une rame de métro

### 3 Résolution parallèle à mémoire distribuée

#### 3.1 Principe et problématique des liaisons *remote*

La résolution parallèle repose sur une décomposition de domaine, répartissant les éléments sur les unités de calcul disponibles en minimisant classiquement la taille des interfaces entre les domaines. Formellement, le système [Eq. 1] devient, pour l'exemple de 3 sous-domaines :

$$(a) \begin{bmatrix} \mathbf{M}_1 & & \\ & \mathbf{M}_2 & \\ & & \mathbf{M}_3 \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{U}}_1 \\ \ddot{\mathbf{U}}_2 \\ \ddot{\mathbf{U}}_3 \end{bmatrix} + \begin{bmatrix} \mathbf{F}_{i1} \\ \mathbf{F}_{i2} \\ \mathbf{F}_{i3} \end{bmatrix} + \mathbf{C}^T \begin{bmatrix} \boldsymbol{\Lambda}_1 \\ \boldsymbol{\Lambda}_2 \\ \boldsymbol{\Lambda}_3 \end{bmatrix} = \begin{bmatrix} \mathbf{F}_{ext1} \\ \mathbf{F}_{ext2} \\ \mathbf{F}_{ext3} \end{bmatrix} \quad [Eq. 3]$$

$$(b) \mathbf{C} \begin{bmatrix} \ddot{\mathbf{U}}_1 \\ \ddot{\mathbf{U}}_2 \\ \ddot{\mathbf{U}}_3 \end{bmatrix} = \mathbf{B}$$

L'accélération parallèle provient tout d'abord de la distribution de l'effort de calcul du vecteur des des forces internes et d'écriture des liaisons d'interaction fluide-structure.

Cependant, la matrice des connexions  $\mathbf{C}$  doit être considérée au niveau global pour le système multi-domaines, car des liaisons peuvent intervenir entre des nœuds appartenant à des sous-domaines différents, par exemple lors de grands déplacements de fragments de structure dans le domaine fluide. On donne à ces liaisons le qualificatif *remote*, pour signifier qu'elles font intervenir des quantités cinématiques non-locales sur les sous-domaines.

La problématique parallèle associée à ces liaisons *remote* est le traitement au niveau global qu'elles requièrent.

Dans le cas peu fréquent où aucune liaison ne couple des degrés de liberté appartenant à des sous-domaines différents, on peut découpler le système [Eq. 3] et le parallélisme est évident. Au contraire, si des couplages existent, il faut mettre en œuvre une résolution faisant intervenir tous les sous-domaines pour le calcul des multiplicateurs de Lagrange.

L'approche existante consiste à identifier dans l'opérateur de condensation global  $\mathbf{H}$  les blocs solubles à l'intérieur de chaque sous-domaine, i. e. correspondant à des groupes de liaisons ne faisant intervenir que des nœuds locaux au sous-domaine concerné. La figure 4 illustre le processus dans le cas élémentaire de 2 sous-domaines, à partir du profil de la matrice des connexions. Sur chaque sous-domaine, on partitionne les degrés de liberté entre ceux impliqués dans des liaisons *remote* (ou des liaisons couplées à ces liaisons *remote*, qui doivent être traitées simultanément) et les autres.

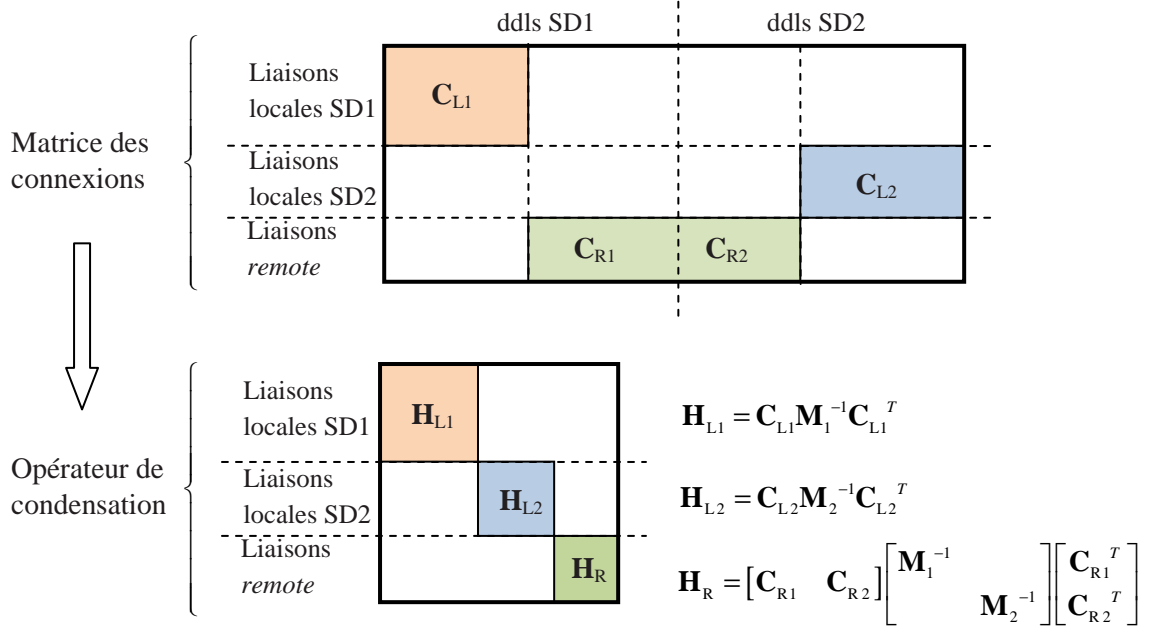


FIG. 4 – Identification des blocs diagonaux dans l'opérateur de condensation

Le bloc global  $\mathbf{H}_R$  est résolu sur un unique processeur, provoquant une perte de parallélisme d'autant plus importante que les couplages cinématiques entre sous-domaines sont nombreux. Ceci peut devenir très dommageable pour l'extensibilité parallèle, à nouveau en présence par exemple de grands déplacements de fragments de structure interagissant avec des éléments fluides initialement distants.

Il est important de souligner que l'assemblage du bloc global  $\mathbf{H}_R$  demande la centralisation des informations des liaisons concernées sur le processeur chargé de la résolution. Une fois ces communications effectuées, il n'est pas pertinent de mettre en œuvre un solveur linéaire distribué, le temps nécessaire à la redistribution de la matrice avant la résolution du système compensant largement l'accélération parallèle du processus de résolution.

### 3.2 Solutions accélératrices

En conservant la même structure parallèle à mémoire distribuée pour le programme, deux traitements palliatifs sont envisagés pour la problématique présentée au paragraphe précédent. Ils sont en cours de mise en œuvre dans le cadre du projet ANR REPDYN [4]. Dans les deux cas, il s'agit de réintégrer du parallélisme dans la résolution du problème intervenant sur les liaisons *remote*.

#### 3.2.1 Utilisation adéquate d'un solveur linéaire parallèle

Comme signalé précédemment, cette option n'est pas judicieuse si l'opérateur est construit de manière centralisée sur un unique processeur. Pour être efficace, un solveur linéaire distribué doit partir d'une matrice dont les termes (par lignes ou individuellement suivant les solveurs) sont déjà répartis sur les ressources disponibles, en nombre équilibré.

Une approche est alors de partager entre tous les processeurs les informations sur les liaisons *remote* et les liaisons qui leur sont couplées, permettant la construction du bloc  $\mathbf{H}_R$  à résoudre au niveau global. Cela représente un surcoût de communications réduit par rapport au transfert actuel vers

une unique destination.

Chaque processeur est alors à même de calculer les termes du bloc  $\mathbf{H}_R$ . Son assemblage redevient ainsi une tâche parallèle et on contrôle la distribution de la matrice du système à résoudre pour tirer parti d'un solveur parallèle, tel que MUMPS [5][6] par exemple.

### 3.2.2 Résolution itérative sans assemblage de l'opérateur global

Même si la construction du bloc global devient parallèle et produit une matrice distribuée, l'extensibilité des solveurs linéaires est souvent limitée, d'autant plus dans le cas présent où la dimension de la matrice est réduite, le nombre de nœuds liés étant souvent inférieur d'un ordre de grandeur ou plus au nombre total de nœuds.

Une stratégie alternative consiste à éviter l'assemblage du bloc global, dans le cadre d'une résolution itérative du problème, avec un algorithme de gradient conjugué par exemple. En reprenant l'exemple du système [Eq. 3], à partir d'un itéré  $k$  du vecteur des multiplicateurs de Lagrange, on obtient les accélérations correspondantes par le système [Eq.3 (a)] :

$$\begin{bmatrix} \ddot{\mathbf{U}}_1^k \\ \ddot{\mathbf{U}}_2^k \\ \ddot{\mathbf{U}}_3^k \end{bmatrix} = \begin{bmatrix} \mathbf{M}_1 & & \\ & \mathbf{M}_2 & \\ & & \mathbf{M}_3 \end{bmatrix}^{-1} \left( \begin{bmatrix} \mathbf{F}_{ext1} \\ \mathbf{F}_{ext2} \\ \mathbf{F}_{ext3} \end{bmatrix} - \begin{bmatrix} \mathbf{F}_{i1} \\ \mathbf{F}_{i2} \\ \mathbf{F}_{i3} \end{bmatrix} - \mathbf{C}^T \begin{bmatrix} \boldsymbol{\Lambda}_1^k \\ \boldsymbol{\Lambda}_1^k \\ \boldsymbol{\Lambda}_1^k \end{bmatrix} \right) \quad [\text{Eq. 4}]$$

On évalue le résidu sur les liaisons cinématiques par :

$$\mathbf{S}^k = \mathbf{C} \begin{bmatrix} \ddot{\mathbf{U}}_1^k \\ \ddot{\mathbf{U}}_2^k \\ \ddot{\mathbf{U}}_3^k \end{bmatrix} - \mathbf{B} \quad [\text{Eq. 5}]$$

Ces deux évaluations sont des étapes parallèles utilisant des opérateurs locaux sur les sous-domaines. Pour l'évaluation du résidu, le vecteur est calculé par blocs, chaque bloc correspondant aux liaisons écrites sur un sous-domaine donné.

Préconditionner l'algorithme est nécessaire pour accélérer la convergence. Un préconditionneur d'intérêt consiste à résoudre les liaisons écrites sur un sous-domaine, impliquant éventuellement des degrés de liberté qu'il ne possède pas, sans tenir compte des couplages possibles avec des liaisons sur les autres sous-domaines (ce qu'assure la construction de l'opérateur global).

En reprenant le cas simplifié de la figure 4, cela revient tout d'abord à séparer dans le bloc des liaisons *remote* de la matrice des connexions les liaisons, i. e. les lignes de la matrice, écrites sur chaque sous-domaine, soit:

$$[\mathbf{C}_{R1} \quad \mathbf{C}_{R2}] = \begin{bmatrix} \mathbf{C}_{R1}^1 & \mathbf{C}_{R2}^1 \\ \mathbf{C}_{R1}^2 & \mathbf{C}_{R2}^2 \end{bmatrix} \quad [\text{Eq. 6}]$$

L'opérateur condensé global s'écrit alors :

$$\mathbf{H}_R = \begin{bmatrix} \mathbf{C}_{R1}^1 & \mathbf{C}_{R2}^1 \\ \mathbf{C}_{R1}^2 & \mathbf{C}_{R2}^2 \end{bmatrix} \begin{bmatrix} \mathbf{M}_1^{-1} & \\ & \mathbf{M}_2^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{C}_{R1}^{1T} & \mathbf{C}_{R1}^{2T} \\ \mathbf{C}_{R2}^{1T} & \mathbf{C}_{R2}^{2T} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_R^1 & \mathbf{H}_{RC}^1 \\ \mathbf{H}_{RC}^2 & \mathbf{H}_R^2 \end{bmatrix} \quad [\text{Eq. 7}]$$

$$\begin{aligned} \text{avec} \quad \mathbf{H}_R^1 &= \mathbf{C}_{R1}^1 \mathbf{M}_1^{-1} \mathbf{C}_{R1}^{1T} + \mathbf{C}_{R2}^1 \mathbf{M}_2^{-1} \mathbf{C}_{R2}^{1T} \\ \mathbf{H}_R^2 &= \mathbf{C}_{R1}^2 \mathbf{M}_1^{-1} \mathbf{C}_{R1}^{2T} + \mathbf{C}_{R2}^2 \mathbf{M}_2^{-1} \mathbf{C}_{R2}^{2T} \\ \mathbf{H}_{RC} &= \mathbf{C}_{R1}^1 \mathbf{M}_1^{-1} \mathbf{C}_{R1}^{2T} + \mathbf{C}_{R2}^1 \mathbf{M}_2^{-1} \mathbf{C}_{R2}^{2T} \end{aligned}$$

Les blocs  $\mathbf{H}_R^1$  et  $\mathbf{H}_R^2$  peuvent être assemblés localement sur les sous-domaines 1 et 2 respectivement, à partir de la seule connaissance des liaisons construites sur ces sous-domaines et de la masse de tous les nœuds impliqués, éventuellement distants.

Ignorer les couplages revient alors à négliger le bloc  $\mathbf{H}_{RC}$  dans la résolution. On obtient ainsi un préconditionneur diagonal par blocs donc chaque bloc peut être traité localement sur un sous-domaine. L'extension à un nombre quelconque de sous-domaines est immédiate.

## 4 Vers un parallélisme hybride

En complément de la recherche d'extensibilité pour la solution du problème sur les connexions cinématiques avec décomposition de domaine, l'évolution des architectures matérielles propose un parallélisme alternatif : un traitement à mémoire distribué avec décomposition de domaines entre des nœuds interconnectés par un réseau à haute performance, couplé à une accélération à mémoire partagée utilisant les cœurs locaux d'un nœud, avec une accélération locale supplémentaire éventuelle sur GPU.

D'un point de vue théorique, cela permet de réduire la concurrence entre la décomposition de domaine et le traitement de connexions cinématique à nombre de cœurs constant, à condition de disposer d'un parallélisme multi-CPU/GPU à mémoire partagée performant.

L'examen des approches de cette nature est à nouveau réalisé pour EUROPLEXUS dans le cadre du projet ANR REPDYN, en collaboration avec l'équipe MOAIS du laboratoire INRIA/LIG. L'équilibrage dynamique de la charge de calcul sur les ressources disponibles au sein d'un nœud est envisagé à l'aide de la librairie KAAPI [7][8], mettant en œuvre des méthodes de partitionnement de graphes et de *work stealing* entre les *threads*.

Un objectif strict est de proposer un parallélisme hybride coopératif entre les ressources couvrant l'intégralité des fonctionnalités du programme. L'originalité est alors de proposer des traitements adaptés à la multiplicité des algorithmes intervenant simultanément dans une simulation couplée, à la différence des démonstrations de performance parallèle sur des applications modèles difficiles à étendre en environnement industriel.

## Références

- [1] F. Casadei. *Fast Transient Fluid-Structure Interaction with Failure and Fragmentation*, 8th World Congress on Computational Mechanics, June 30-July 5, 2008.
- [2] V. Faucher, S. Kokh. *Explosive Fluid-Structure Interaction using Multi-Component Flows with Anti-Dissipation*. IV European Conference on Computational Mechanics, May 16-21, 2010.
- [3] F. Casadei, J.P. Halleux, A. Saha, F. Chille. *Transient Fluid-Structure Interaction Algorithm for Large Industrial Applications*, Comp. Meth. in Appl. Mech. and Engrg., 2001.
- [4] <http://www.repdyn.fr>
- [5] P.R. Amestoy, I.S. Duff & J.-Y. L'Excellent. *Multifrontal parallel distributed symmetric and unsymmetric solvers*, ENSEEIHT-IRIT Technical Report, 1998. Revised version appeared in Comp. Meth. in Appl. Mech. Eng., 184, 501-520 (2000).
- [6] <http://graal.ens-lyon.fr/MUMPS/>
- [7] T. Gautier, X. Besseron & L. Pigeon. *KAAPI : a thread scheduling runtime system for data flow computations on cluster of multi-processors*. In Parallel Symbolic Computation'07 (PASCO'07), number 15-23, London, Ontario, Canada, 2007.
- [8] <http://kaapi.gforge.inria.fr/>